Time Complexity of Matrix Transpose Algorithm using Identity Matrix as Reference Matrix

Sanil Shanker KP¹ , Mohammed Shameer MC²

¹²Dept. of Computer Science
Farook College, Kozhikode, India.

Abstract- This paper presents the time complexity of matrix transpose algorithm using identity matrix as reference matrix. We computed the time complexity of the algorithm as $O(mn)$.

Keywords: Identity matrix, Reference matrix, Sanil’s Matrix Transpose.

I. INTRODUCTION

Transpose of the matrix can be obtained by combining the characteristics of logical AND ($∧$) with logical OR ($∨$) operations [1, 2]. In Sanil’s matrix transpose algorithm, the identity matrix acts as the kernel of the transformation [3]. For example, let the matrix $A(3 \times 4)$ be

\[
\begin{pmatrix}
17 & 2 & 13 & 7 \\
41 & 11 & 29 & 19 \\
19 & 3 & 23 & 11
\end{pmatrix}
\]

The transformation can be computed as:

\[
\begin{pmatrix}
17 & 2 & 13 & 7 \\
41 & 11 & 29 & 19 \\
19 & 3 & 23 & 11
\end{pmatrix} ∧ I_3
\]

Output: $A^T(4 \times 3)$

II. TIME COMPLEXITY

Let $A_{m \times n}$ and $B_{m \times m}$ be the input matrix of order $(m \times n)$ and the reference matrix of order $(m \times m)$ respectively. The value of $c_{mn}$ can be computed from the Figure- 1, as $c_{11} := (a_{11} \cdot b_{11}) + (a_{21} \cdot b_{21}) + (a_{31} \cdot b_{31})$.

\[
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & a_{14} \\
a_{21} & a_{22} & a_{23} & a_{24} \\
a_{31} & a_{32} & a_{33} & a_{34}
\end{pmatrix}
\begin{pmatrix}
b_{11} & b_{12} & b_{13} \\
b_{21} & b_{22} & b_{23} \\
b_{31} & b_{32} & b_{33}
\end{pmatrix}
\]

To compute one cell value, there exists ‘$m$’ multiplications and ‘$m$-1’ additions. For the transformation, $c_{mn} \leftarrow a_{mn}$, the computational time is $O(m)$. If there exists ‘$m$’ rows, time will be $O(m) + O(m) + \ldots \ldots \ldots m$ times = $O(m^2)$. For ‘$n$’ columns, the computational time is $O(mn^2)$.

In the case of identity matrix as reference matrix, $(a_{i} = m, j = n \cdot I_i = m, j = m)$ exists and other will be zero (Figure- 2) [2]. This implies the time for one multiplication operation will be O(1). If there exists ‘$m$’ rows, time will be $O(1) + \ldots \ldots m$ times = $O(m)$. In general, for ‘$n$’ columns, time = $O(mn)$. 

www.ijcsit.com
III. SUMMARY

The computational time of matrix transpose algorithm using identity matrix as reference matrix is $O(mn)$.

Suppose, if the given matrix is a square matrix, the running time will be $O(n^2)$.

REFERENCES

